

Работа с модулем sqlite3 на языке программирования Python

Андрюенко Иван Сергеевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье описывается процесс создания таблицы, добавления, проверки и вывода данных в базу данных sqlite на языке программирования Python. Для работы с базой данных sqlite использовалась библиотека sqlite3. В итоге была создана база данных, содержащая таблицу с номерами клиентов.

Ключевые слова: SQLite, Python, база данных, таблица

Working with the sqlite3 module in the Python programming language

Andrienko Ivan Sergeevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article describes the process of creating a table, adding, checking, and outputting data to a sqlite database. To work with the sqlite database, the sqlite3 library was used. As a result, a database containing a table with customer numbers was created.

Keywords: Sqlite, Python, database, table

1 Введение

1.1 Актуальность

Базы данных необходимы для любого предприятия. Они значительно упрощают работу с информацией, ее обработку, изменение и удаление. Использование баз данных на предприятиях весьма актуально на сегодняшний день. Актуальностью этой темы является то, чтобы упростить доступ к информации для пользователей, а также эффективно управлять и организовать деятельность автосервиса.

1.2 Обзор исследований

В своей работе А.А. Когутенко, С.В. Плотникова рассмотрели этапы создания кроссплатформенной рабочей среды с облачным хранилищем данных. Изучены особенности организации облачных хранилищ данных. Рассмотрено применение и предназначение, достоинства и недостатки основных веб-серверов, серверных языков, серверных систем управления базами данных (MySQL, SQLite, PostgreSQL) с целью выбора оптимальных

инструментов для разработки и обеспечения работоспособности среды [1]. Л.А. Суходолова проанализировала преимущества и недостатки SQLite. Также была решена поставленная задача [2]. А.В. Химич рассмотрел основные виды баз данных, описал основные возможности реализации баз данных в языке программирования Python. Обоснован выбор модуля SQLite для создания приложений на языке программирования Python [3]. В своей работе А.И. Павлова Разработала программу для оптимального выбора сельскохозяйственной техники на языке высокого уровня Python. В программе реализованы авторизация пользователя, работа пользователя с базой данных сельскохозяйственной техники. При этом использована база данных SQLite, позволяющая хранить и обрабатывать данные с применением облачных не клиент-серверных технологий [4]. Э.В. Гарайс, М.Д. Винников в своей научной статье рассмотрели пример построения веб-сервера для мобильного приложения для мотоциклистов. В качестве средств реализации используются скриптовый язык Python и фреймворк для этого языка - Flask. Данные хранятся в базе данных SQLite, управляемой с помощью библиотеки Flask SQLAlchemy. [5].

1.3 Цель исследования

Цель исследования – создать таблицу, добавить, проверить и вывести данные в базе данных sqlite на языке программирования Python.

2 Материалы и методы

Для работы с базой данных sqlite использовалась библиотека sqlite3. Для написания кода использовалась среда программирования PyCharm.

3 Результаты и обсуждения

SQLite — это библиотека, которая предоставляет облегченную базу данных на основе диска, не требующая отдельного серверного процесса, и позволяющая получить доступ к базе данных с использованием нестандартного варианта языка запросов SQL. Некоторые приложения могут использовать SQLite для внутреннего хранения данных.

Перед работой с базой данных, необходимо установить библиотеку sqlite3. Для этого пропишем в терминал строку «pip install sqlite3».

Теперь настроим подключение к базе данных. Для начала импортируем библиотеку. Затем сразу подключаем базу данных «list.db». Если такой базы данных в папке проекта нет, то она автоматически создастся. Создаем переменную sql для работы с базой данных, за это отвечает «cursor» (рис. 1).

```
1 import sqlite3
2
3 db = sqlite3.connect('list.db')
4 sql = db.cursor()
```

Рисунок 1 – Подключение к БД

Далее создаем таблицу «users» в созданной базе данных. Если использовать оператор «CREATE TABLE», то при перезапуске программы будет выдана ошибка, что такая таблица уже существует. Для этого дописываем к нему «IF NOT EXISTS». В таблице создадим два столбца: number с типом данных BIGINT и password с типом данных TEXT. С помощью «commit» подтверждаем действия (рис. 2).

```
6 sql.execute("""CREATE TABLE IF NOT EXISTS users(  
7     number BIGINT,  
8     password TEXT  
9 )""")  
10 db.commit()
```

Рисунок 2 – Создание таблицы

Теперь создаем две переменные с помощью которых пользователь будет вносить в базу данных (рис.3).

```
12 user_number = input('Номер: ')  
13 user_password = input('Пароль: ')
```

Рисунок 3 – Создание переменных

Создадим проверку от повторяющихся логинов, чтобы пользователи не могли иметь одни и те же логины. Выбираем столбец number из таблицы и проверяем его. Если проверка будет провалена, будет выдано сообщение о регистрации пользователя с таким логином. В скобках «VALUES» рекомендуется изначально прописать вопросы, защищающие от sql-инъекции. С помощью метода «execute» добавим в таблицу введенные данные пользователя (рис. 4).

```
15 sql.execute("SELECT number FROM users")  
16 if sql.fetchone() is None:  
17     sql.execute("INSERT INTO users VALUES (?, ?)", (user_number, user_password))  
18     db.commit()  
19  
20     print('Пользователь зарегистрирован')  
21 else:  
22     print('Данный пользователь уже зарегистрирован')
```

Рисунок 4 – Создание проверки и сохранения данных в таблицу

Теперь выведем данные из таблицы, с помощью цикла for. Выбираем все столбцы в таблице users с помощью запроса «*» и выводим данные (рис. 5).

```
24     for value in sql.execute("SELECT * FROM users"):
25         print(value)
```

Рисунок 5 – Вывод данных


Рассмотрим, как удалить созданную таблицу используя оператор «drop».

Для этого пропишем код (рис. 6).

```
27     def sql_fetch(db):
28
29         sql.execute('DROP table if exists users')
30         db.commit()
31
32     sql_fetch(db)
```

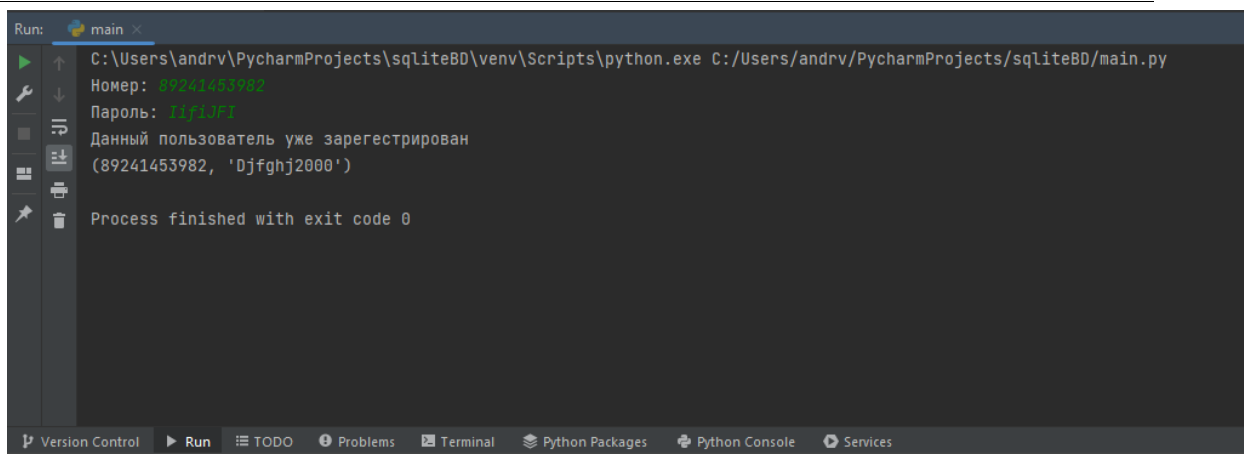
Рисунок 6 – Удаление таблицы

Теперь проверим созданную программу. При её запуске видим запрос логина и пароля пользователя. При успешной регистрации получаем сообщение об успешном вводе (рис. 7), а при повторе регистрации с ранее введенным логином получаем предупреждение (рис. 8). Так же ниже прописана строка с выведенными данными с базы данных.



```
Run: main x
C:\Users\andrv\PycharmProjects\sqliteBD\venv\Scripts\python.exe C:/Users/andrv/PycharmProjects/sqliteBD/main.py
Номер: 89241453982
Пароль: Dfghj2000
Пользователь зарегистрирован
(89241453982, 'Dfghj2000')
Process finished with exit code 0
```

Рисунок 7 – Успешная регистрация



```
Run: main x
C:\Users\andrv\PycharmProjects\sqliteBD\venv\Scripts\python.exe C:/Users/andrv/PycharmProjects/sqliteBD/main.py
Номер: 89241453982
Пароль: 11f1jft
Данный пользователь уже зарегистрирован
(89241453982, 'Djfgjhj2000')
Process finished with exit code 0
```

Рисунок 8 – Предупреждение о повторе логина

Выводы

В данной работе была создана база данных sqlite с таблицей на языке программирования Python. В таблицу были добавлены данные и выведены из неё в консоль.

Библиографический список

1. Когутенко А.А., Плотникова С.В. Исследование организации кроссплатформенной рабочей среды с облачным хранилищем данных. // Современные информационные технологии в образовании и научных исследованиях (СИТОНИ-2017). Материалы V Международной научно-технической конференции. 2018. С. 328-332.
2. Суходолова Л.А. Обзор SQLite в Python для разработки системы подбора косметической продукции по фотографии клиента. // Россия молодая. Сборник материалов XIII Всероссийской научно-практической конференции с международным участием. Редколлегия: К.С. Костиков (отв. ред.) [и др.]. Кемерово, 2021. С. 31467.1-31467.3.
3. Химич А.В. К вопросу о реализации баз данных в языке программирования Python // Ученые записки Брянского государственного университета. 2020. № 2 (18). С. 33-35.
4. Павлова А.И. Разработка программы для оптимального подбора сельскохозяйственной техники // Поколение будущего: Взгляд молодых ученых-2020. Сборник научных статей 9-й Международной молодежной научной конференции, в 5-х томах. Курск, 2020. С. 415-418.
5. Гарайс Э.В., Винников М.Д. Разработка веб-сервера для мобильного приложения с использованием Python и Flask. // Современные научные взгляды в эпоху глобальных трансформаций: проблемы, новые векторы развития. Материалы XLII Всероссийской научно-практической конференции. Ростов-на-Дону, 2021. С. 124-126.